Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2005 Proceedings

Americas Conference on Information Systems (AMCIS)

2005

A Bayes' Theorem Based Approach for the Selection of Best Pruned Tree

Xun Li
University of Kentucky, xli@uky.edu

A Bayes' Theorem Based Approach for the Selection of Best Pruned Tree

Xun Li
Decision Science and Information Systems Department
Gatton College Of Business & Economics
University of Kentucky
e-mail: xli@uky.edu

Abstract

Decision tree pruning is critical for the construction of good decision trees. The most popular and widely used method among various pruning methods is cost-complexity pruning, whose implementation requires a training dataset to develop a full tree and a validation dataset to prune the tree. However, different pruned trees are found to be produced when the original dataset are randomly partitioned into different training and validation datasets. Which pruned tree is the best? This paper presents an approach derived from Bayes' theorem to select the best pruned tree from a group of pruned trees produced by cost-complexity pruning method. The results of an experimental study indicate that the proposed approach works satisfactorily to find the best pruned tree in terms of classification accuracy and performance stability.

Keywords

Selection of best pruned tree, Bayes' theorem

INTRODUCTION

Decision trees are a widely used technique to build classification or regression models. Decision trees are attractive to analysis due to their simplicity and ease in model interpretation (Quinlan, 1993). They are particularly suited for data mining and can be constructed relatively fast compared to other classification methods (Mehta, Rissanen, and Agarwal, 1996). In addition, classification trees can be easily converted to SQL statements that can be used to access databases efficiently (Shafer, Metha and Agarwal, 1996).

Decision tree induction is usually a two-phase process (Breinman, Friedman, Olshen and Stone, 1984; Quinlan, 1993). First a tree reflecting the given sample is constructed by using decision-tree algorithm. The algorithm begins with the entire set of data, splits the data into two or more subsets based on the values of one or more attributes, and then repeatedly splits each subset into finer subsets until the split size reaches an appropriate level. If no noise prevails, the accuracy of the tree is perfect on the training examples that were used to build the tree. In practice, however, "over-fitting" always occurs; in addition to the general trends of the data, the resulting decision tree encodes the peculiarities of the training data, which makes it a poor predictor of instances that are not in training dataset. Therefore, in the second phase of induction, the decision tree is pruned in order to generalize well on a test set.

The majority of studies in decision tree pruning focus on pruning methods, such as cost- complexity pruning (Breiman et al. 1984), reduced-error pruning (Quinlan, 1987), pessimistic-error pruning (Quinlan 1987, 1993), minimum-error pruning (Niblett, 1986), critical value pruning (Mingers, 1987), a bootstrap-based pruning method (Crawford, 1989) and so on.

Among various pruning methods, cost-complexity pruning (CCP) is perhaps the most popular and widely used method. Many systems such as CART, S-Plus, OC1 implement CCP. To use CCP, two data sets are required. The training dataset is used to build the full tree, while the validation dataset is for pruning the tree. The resulting pruned tree will be regarded as the final decision tree model for the original dataset. In my experience with CCP based decision tree systems, however, a problem never addressed is found: for most of the time, a different pruned tree is produced each time the original large dataset is partitioned randomly into a different training data set and a validation data set. Which pruned tree is the best among all the possible available pruned trees?

The objective of this paper is to seek a method to find the best pruned tree from all the possible pruned trees produced by CCP on different partition results of the same dataset. This paper starts with a discussion of current methods to select best tree. There follows a new approach for best tree selection, which is derived from Bayesian theorem. Then the method is validated in an empirical experiment. This paper concludes with what can be done in future research.

CURRENT EVALUATION CRITERIA FOR BEST TREE SELECTION

In literature, there are basically two approaches to tree model selection. One is cost ratio, and another is total misclassification error rate.

In binomial classification tree models, two types of misclassification errors are encountered: Type I and Type II. A Type I error occurs when a class I instance is classified as class II and a Type II error occurs when a class II instance is classified as class I. If the cost of Type II misclassification (C_{II}) and cost of Type I misclassification (C_I) are known, and Type II errors are more severe in terms of cost, Tachi M. Khoshgoftaar and Seliya (2003) suggests that the final model selection would depend on the modeling cost ratio (C_{II}/C_I). However, the major difficulty with this approach is the costs are unknown at the time of modeling. Therefore, this is not an appropriate approach for general tree model selection.

The second approach is using the total misclassification error rate (TMER): the model with the smallest total misclassification rate is the best. However, the next experiment results show that this approach is not viable either.

To implement CCP, a software called XLMiner is used, which can be bought at: http://www.resample.com. The experimental dataset, German Credit, is provided by the software. This dataset has 1000 records and 30 variables. For ten times, 500 records out of 1000 are randomly selected as the training dataset for growing the full tree, while the rest 500 as the validation dataset for pruning. The 10 resulting pruned trees are then applied to the original dataset to get TMER for comparison. The results are shown in Table1.

	Total Misclassification Error Rate	
# of Pruned tree	Validation Dataset	Original Dataset
9	28.40%	24.20%
1	27.40%	25.30%
8	25.60%	25.50%
7	25.00%	25.60%
3	28.60%	25.80%
4	29.40%	26.50%
2	29.00%	26.90%
5	29.40%	26.90%
6	27.40%	27.10%
10	28.20%	30.00%

Table1: Total misclassification error rates for validation datasets vs. for original dataset

It is obvious that the ability of pruned tree in catching the generality of datasets is not consistent. The pruned tree with lower total misclassification error rate on the validation dataset doesn't always perform better on the original dataset than the one with higher total misclassification error rate on its validation dataset. For example, pruned tree 7 has the lowest total misclassification error rate (25%) among the ten pruned trees, but its performance on original dataset (Total misclassification error rate is 25.6%) is worse than that of pruned tree 9, 1 or 8, with 28.4%, 27.4% and 25.6% error rates on validation datasets and 24.2%, 25.3% and 25.5% error rates on original dataset respectively.

Therefore, this paper is motivated to find another evaluation method to find the best model. My approach relies on Bayes' theorem, a basic principle of logic. The details of the approach will be addressed in next section.

A NEW APPROACH: EVALUATE CLASSIFICATION ACCURACY AND PERFORMANCE STABILITYA

For a binomial classification problem, even without any other help, it is still reasonable to say that the error rate of assigning any instance incorrectly is at most 50%, because there are only two options for the classification of one instance, either class 1 or class 2. In another word, for binomial classification, the random probability of correct classification or incorrect classification is 50%, without other information known.

Now, with the new information or evidence a pruned tree model brings about the dataset, how do we revise the belief in the probability of accurate classification? My approach relies on Bayes' theorem, which indicates how a rational evaluator should adjust a probability assessment in light of new evidence.

Suppose we must assess the probability of two alternative hypotheses:

H_1: The classification is correct;
H_2: The classification is incorrect.

The evidence to be evaluated is what a pruned tree model reveals about the dataset, which is called E. We will get the following formula based on the conventional expression of Bayes' theorem:

$$\underbrace{\left\{\frac{P(H_1)}{P(H_2)}\right\}}_{\text{Priors}}\underbrace{\left\{\frac{P(E|H_1)}{P(E|H_2)}\right\}}_{\substack{\text{Likelihood}\\ \text{Ratio}}}=\underbrace{\left\{\frac{P(H_1|E)}{P(H_2|E)}\right\}}_{\text{posteriors}} \qquad (1)$$

Bayes' theorem describes the relationship between tree components: the prior odds, the likelihood ratio and the posterior odds. The prior odds reflect the evaluator's assessment of the odds that a hypothesis is true before the receipt of new evidence. The posterior odds reflect the evaluator's belief in the odds that the hypothesis is true after receipt of new evidence. The likelihood ratio is defined as the ratio of the probability of some piece of evidence E given hypothesis H_1 to the probability of E given a competing hypothesis H_2. In forensic context, likelihood ratios are an accepted concept for weighing evidence (F. Taroni, Biedermann, Garbolino and Aitken, 2004).

Suppose, for example, that we initially think that there is a 60% chance that an instance can be classified correctly. In terms of Eq 1, $P(H_1)=0.6$ and $P(H_2)=0.4$. Therefore, the prior odds would be 1.5. Suppose further that we thinks the chance that a given pruned tree model classifies instances correctly is 80%, and the chance of misclassification is 20%. Accordingly, the likelihood ratio is $P(E|H_1)/ P(E|H_2) =0.8/0.2=4$. The posterior odds would be 1.5 * 4 = 6. In other words, we should now believe that instances will be classified correctly 6 times more likely than misclassified.

The conclusion can be restated as a probability by simply converting the posterior odds to a probability using the formula: Probability=Odds/(Odds + 1). Thus, we can say that the probability that correct classification is 6/7=0.857. In another word, if we believe the given pruned tree model is 6 times more likely to classify the instances accurately than inaccurately, we can revise our estimated probability of correct classification from 0.6 to 0.857.

As we discussed above, without any other information, the probability of correct classification or incorrect classification is 50% for binomial classification. That means the prior odds of classification hypothesis equal to 1 (50%/50%).

$$\frac{P(H_1)}{P(H_2)} = 1$$

Therefore, to get the posterior odds, we still have the likelihood ratio in short.

In the conventional expression of Bayes' theorem, the likelihood ratio takes into account all variables that affect the value of the evidence. In terms of classification problem, the identification of factors leading the pruned tree to misclassification will help determine the value of likelihood ratio.

In an effort to illustrate the role that error may play in determining the value of DNA evidence, Thompson et al. (2004) prove that the likelihood ratio can be expanded to show the separate effect of the random match probability and the false positive probability on the value of a reported DNA match.

$$\frac{P(R|S_1)}{P(R|S_2)} = \frac{P(R|M_1)}{P(R|M_1) * P(M_1|\ S_2) + P(R|M_2) * P(M_2|\ S_2)} \qquad (2)$$

S_1: The specimen came from a suspect;
S_2: The specimen did not come from a suspect.
M_1: The suspect and the specimen have matching DNA profiles;
M_2: The suspect and the specimen do not have matching DNA profiles.
R: The laboratory report of a DNA match between the suspect's profile and the profile of the sample.

In the above expanded version of the likelihood ratio, the term $P(R|M_1)$ is the probability that the laboratory will report a match if the suspect and the specimen have matching DNA profiles. The term $P(M_1|\ H_2)$ is the probability of a coincidental match. A coincidental match occurs when two different people have the same DNA profile. $P(R|M_2)$ is the false positive probability. A false positive occurs when a laboratory erroneously reports a DNA match between two samples that actually have different profiles.

Likewise, Equation (2) also applies to classification problem with some modification and is rewritten as the following:

$$\frac{P(E|H_1)}{P(E|H_2)} = \frac{P(E|M_1)}{P(E|M_1) * P(M_1| H_2) + P(E|M_2) * P(M_2| H_2)} \qquad (3)$$

H_1: The classification of an instance is correct;
H_2: The classification of an instance is incorrect.
M_1: The class of an instance matches its classification;
M_2: The class of an instance does not match its classification;
E: The classification of pruned tree model.

The term $P(E|M_1)$ is the probability that the pruned tree model assigns an instance correctly if the class of that instance matches its classification. The value of $P(M_1| H_2)$ is 0 because the probability of a coincidental match won't occur for classification problem where each instance can only belong to one class. Because M_1 and M_2 are mutually exclusive, $P(M_2| H_2)$ is the complement of $P(M_1| H_2)$ an its value is 1. Finally, the term $P(E|M_2)$ obviously corresponds to the misclassification error rate of the pruned tree. Substituting terms, Equation (3) can be restated as the following:

$$\frac{P(E|H_1)}{P(E|H_2)} \quad \frac{P(E|M_1)}{P(E|M_2)} \qquad (4)$$

Now, the posteriors odds will be determined by two elements, the correct classification rate and the misclassification error rate of a given pruned tree. If the posterior odds are restated in probability, then the higher the probability, the more accurate the pruned tree classifies the dataset. The best pruned tree among a group of available pruned trees will be the one with the highest probability of accurate classification.

However, what if the revised probabilities of accurate classification are the same for several trees? The next criteria my approach adopts is the stability of performance, because the pruned tree that performs more consistently on both validation dataset and original dataset is the one with better ability to capture the generality of the mother dataset. How to express performance stability in a quantitative way? It is obvious that a pruned tree will show similar probabilities of classifying class1 instances as class1 and class2 instances as class2 when it is applied to different datasets, if it can capture the generality of the dataset. Therefore, a comparison of probabilities of assigning class1 instances correctly by a pruned tree on both its validation dataset and the original dataset can help judge the performance stability of that pruned tree. We still can use the Bayes' theorem to calculate the probabilities. Suppose we have two hypotheses:

C_1: An instance belongs to class 1;
C_2: An instance belongs to class 2.

The prior odds of the classification hypotheses are:

$$\frac{P(C_1)}{P(C_2)} = \frac{(\#\text{ of instances in class 1})}{(\#\text{ of instances in class 2})}$$

The posterior odds tell us the adjusted probability of assigning an instance to class 1 as opposed to class 2, given the new information provided by a given pruned tree:

$$\frac{P(C_1|E)}{P(C_2|E)} = \frac{P(C_1)\, P(E|C_1)}{P(C_2)\, P(E|C_2)} \qquad (5)$$

The term $P(E|C_1)$ is the probability of an instance being assigned to class 1 if that instance belongs to class 1, while $P(E|C_2)$ is the probability of an instance being assigned to class 2 if that instance belongs to class 2. Two sets of posterior odds are produced when Equation (5) is used to evaluate a given pruned tree's performance of assigning class 1 instances on its validation dataset and the original dataset. If the evaluated pruned tree has the ability to capture the generality of the original dataset, as discussed above, the probability resulted from the posterior odds calculated on the original dataset should be very close to that on the validation dataset. In another word, the closer the two probabilities from the two sets of posterior odds, the more stable the pruned tree performs.

For example, if the same probability occurs to pruned tree 1 and pruned tree 2 after Equation (4) is used to weigh their values on classification accuracy on the original dataset, Equation (5) is used to evaluate the stability of their performance. For pruned tree 1, suppose the probabilities derived from the posterior odds calculated on the validation dataset and on the original dataset are 80% and 81%. The difference of the two probabilities is 1%. For pruned tree 2 with 78% and 80% on validation and the original dataset respectively, the difference is 2%. Therefore, tree 1 is selected over tree 2.

EXPERIMENTAL EVALUATION

To verify our approach to find the best pruned tree, another 20 pruned trees are produced by partitioning the German Credit dataset in different ways. The tree producing process is the same as that described for the ten pruned trees in Table1. Among the 20 trees, five pruned trees are produced on training datasets and validation datasets, each with 300 records respectively; another five trees come from datasets with 200 records; the final ten pruned trees are based on datasets with 250 records. The results are shown in Appendix1.

In the 30 trees, some trees are found to have the same structure and some just simply assign all instances to one class. After getting rid of those trees, I have 18 trees left. Each tree is then applied to the original dataset to get the confusion table, which helps the calculation of the elements determining the posterior odds in Equation (4).

For example, pruned tree 4 has the following confusion table:

Classification Confusion Matrix		
	Predicted Class	
Actual Class	1	0
1	626	74
0	172	128

From the above confusion table, it is easy to calculate that $P(E|M_2)$ =(246/1000)=0.246, and $P(E|M_1)$=(626+128)/1000=0.754. Then Equation (4) can be used to obtain the posterior odds, which can be restated as the probability of accurate classification for that given pruned tree. Table2 shows the 18 trees, which are ranked based on the criteria, the probability of accurate classification on the original dataset.

# of Pruned tree	Prior Odds	$P(E\|M_2)$	$P(E\|M_1)$	posterior odds	Probability (Accurate Classification)
27	1	0.242	0.758	3.1322	0.758
4	1	0.246	0.754	3.0650	0.754
8	1	0.25	0.75	3.0000	0.750
20	1	0.253	0.747	2.9526	0.747
25	1	0.253	0.747	2.9526	0.747
3	1	0.254	0.746	2.9370	0.746
29	1	0.255	0.745	2.9216	0.745
10	1	0.256	0.744	2.9063	0.744
26	1	0.256	0.744	2.9063	0.744
14	1	0.258	0.742	2.8760	0.742
22	1	0.258	0.742	2.8760	0.742
13	1	0.259	0.741	2.8610	0.741
11	1	0.263	0.737	2.8023	0.737
23	1	0.265	0.735	2.7736	0.735
21	1	0.269	0.731	2.7175	0.731
5	1	0.271	0.729	2.6900	0.729
30	1	0.271	0.729	2.6900	0.729
19	1	0.276	0.724	2.6232	0.724

Table2: Pruned trees and their probability of accurate classification on the original dataset

It's fortunate that the highest probability of accurate classification has no overlapped value and pruned tree 27 can be declared the best in terms of classification accuracy among the 30 pruned trees produced.

It is also noted that overlapped probability exists between pruned tree 20 and 25, pruned tree 10 and 26, pruned tree 14 and 22 and pruned tree 5 and 30. To rank these trees, Equation (5) is used. Table3 shows the results.

	Original Dataset			Validation Dataset			
# of Pruned tree	Prior Odds	posterior odds	Probability (class 1 Classification)	Prior Odds	posterior odds	Probability (class 1 Classification)	Difference in Probability
25	2.3333	5.6696	0.8501	2.2258	5.7222	0.8512	0.0012
20	2.3333	6.3960	0.8648	2.2609	4.7500	0.8261	0.0387
26	2.3333	6.1538	0.8602	2.4722	6.8125	0.8720	0.0118
10	2.3333	4.7231	0.8253	2.3784	3.3478	0.7700	0.0553
22	2.3333	5.0325	0.8342	2.2468	5.4909	0.8459	0.0117
14	2.3333	3.9799	0.7992	2.7313	4.4324	0.8159	0.0167
30	2.3333	7.8902	0.8875	2.3784	9.6765	0.9063	0.0188
5	2.3333	4.8320	0.8285	2.4483	4.0000	0.8000	0.0285

Table3: Performance stability comparisons

Here is an example about how to use Equation (5) to get values in Table3. For pruned tree 30, two confusion tables are developed on original dataset and its validation dataset respectively.

Classification Confusion Matrix			
	Predicted Class		
Actual Class	1	0	
1	329	23	
0	114	34	
Error Report			
Class	# Cases	# Errors	% Error
1	352	23	6.53
0	148	114	77.03
Overall	500	137	27.40

On validation dataset

Classification Confusion Matrix			
	Predicted Class		
Actual Class	1	0	
1	647	53	
0	218	82	
Error Report			
Class	# Cases	# Errors	% Error
1	700	53	7.57
0	300	218	72.67
Overall	1000	271	27.10

On original dataset

Take validation dataset as example:

$$P(C_1) / P(C_2) = 352/148 = 2.3784$$
$$P(E|C_1) = 329/352 = 0.9347$$
$$P(E|C_2) = 34/148 = 0.2297$$
$$P(C_1|E) / P(C_2|E) = (P(C_1) / P(C_2)) * (P(E|C_1) / P(E|C_2)) = 9.6765$$
$$\text{Probability} = 9.6765 / (9.6765 + 1) = 0.9063$$

Likewise, we can get the probability on the original dataset, which is 0.8875. Therefore, the difference is 0.0188.

The pruned trees in Table2 are re-ranked to reflect the changes after performance stability is considered for trees with the same probability of classification accuracy. In order to validate the ranking accuracy, the original dataset is partitioned randomly to get another 9 datasets ranging from 100 to 900 records, with an increment of 100 records each time. Then each pruned tree is applied to the 9 datasets to get 9 misclassification error rates. An average error rate is calculated based on the 9 error rates plus the one on the original dataset. Table4 shows the final ranking of the 18 pruned trees along with the average error rate on ten datasets.

# of Pruned tree	Prior Odds	$P(E\|M_2)$	$P(E\|M_1)$	posterior odds	Probability (Accurate Classification)	Average Error Rate
27	1	0.242	0.758	3.1322	0.7580	0.2355
4	1	0.246	0.754	3.0650	0.754	0.2490
8	1	0.25	0.75	3.0000	0.7500	0.2481
25	1	0.253	0.747	2.9526	0.747	0.2478
20	1	0.253	0.747	2.9526	0.747	0.2509
3	1	0.254	0.746	2.9370	0.7460	0.2533
29	1	0.255	0.745	2.9216	0.745	0.2490
10	1	0.256	0.744	2.9063	0.744	0.2561
26	1	0.256	0.744	2.9063	0.7440	0.2569
22	1	0.258	0.742	2.8760	0.7420	0.2568
14	1	0.258	0.742	2.8760	0.7420	0.2578
13	1	0.259	0.741	2.8610	0.7410	0.2581
11	1	0.263	0.737	2.8023	0.7370	0.2631
23	1	0.265	0.735	2.7736	0.7350	0.2613
21	1	0.269	0.731	2.7175	0.7310	0.2652
30	1	0.271	0.729	2.6900	0.7290	0.2670
5	1	0.271	0.729	2.6900	0.7290	0.2752
19	1	0.276	0.724	2.6232	0.7240	0.2735

Table4: Final rank of pruned trees and validation against average error rate of ten datasets

It is very obvious in Table4 that the relatively higher probability corresponds to the relatively lower average error rate most of the time. Specifically, for this experiment, only 3 out of 18 pruned trees (pruned tree 4, 8, 29) are in the wrong ranking positions.

Based on the experimental results in Table4, it is fair to conclude that, in this study, the approach based on Bayes' theorem to select the best pruned tree among a group of available pruned trees performs successfully.

CONCLUSION

This paper has presented a new approach that is derived from Bayes' theorem to select the best pruned tree among a group of available pruned trees produced by cost-complexity pruning algorithm on the different partition results from the same original dataset. The results of the experimental study indicate that this new approach works satisfactorily. However, more empirical studies are needed in order to make definitive conclusions regarding the effectiveness of this approach on other datasets. In addition, the average error rates are calculated only on 10 datasets. To get a more accurate validation criterion, the average error rates should be calculated on at least 30 datasets.

As observed earlier, the proposed approach is only tested on binary trees. It will be interesting, to investigate the possible extension of this approach to higher-order trees in future research.

There also exist many research opportunities involving the opposed approach. One example is a comparative study on the performance of different pruning algorithms through the usage of this approach.

Besides classification accuracy and performance stability, misclassification cost is another important issue in the selection of quality decision trees. It is sometimes more expensive to misclassify a class 2 instance as class 1 than to misclassify a class 1 instance as class 2. How to tie up the tree criteria in one approach to the selection of the premium pruned tree is an interesting and important topic for future research.

REFERENCES

1. Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J. (1984), Classification and Regression Trees, Wadsworth, Belmont, CA.
2. Crawford, S. L. (1989), Extensions to the CART algorithm, *International Journal Man-Machine Studies*, 31, 197–217.
3. Khoshgoftaar, Tachi M. and Seliya, Naeem (2003), Software Quality Classification Modeling Using the SPRINT Decision Tree Algorithm, *International Journal on Artificial Intelligence Tools*, 12, 3, 207-225.
4. Mehta, M., Rissanen, J., and Agarwal, R. (1996), SLIQ: A fast scalable classifier for data mining, *Proceedings of the Fifth International Conference on Extending Database Technology*, March, Avignon, France.
5. Mingers, J. (1987), Expert systems—rule induction with statistical data, *Journal of Operational Research Society,* 38, 39–47.
6. Niblett, T. (1986), Constructing decision trees in noisy domains, *Progress in Machine Learning*, Sigma Press, England.
7. Quinlan, J. R. (1987), Simplifying decision trees, *International J. Man-Machine Studies*, 27, 221–234, 1987.
8. Quinlan, J.R. (1993) C4.5: Programs for Machine Learning, *Morgan Kaufman Series in Machine Learning*.
9. Shafer, J., Mehta, M., and Agarwal, R. (1996), Sprint: A scalable parallel classifier for data mining, *Proceedings of the Twenty Second VLDB Conference*, Bombay, India.
10. Taroni, F., Biedermann, A., Garbolino, P., Aitken, C.GG. (2004), A general approach to Bayesian networks for the interpretation of evidence, *Forensic Science International*, 139, 5-16.
11. Thompson, William C., Taroni, Fanco, Aitken, Colin G.G. (2003), How the probability of a false positive affects the value of DNA evidence, *Journal of Forensic Science*, 48, 1.

APPENDIX

1. 30 Pruned Trees and Their Misclassification Error Rates on Validation Datasets

Dataset: 200

Pruned Tree	Misclassification Error rate
1	0.28
2	0.29
3	0.23
4	0.21
5	0.25

Dataset: 250

6	0.28
7	0.312
8	0.268
9	0.268
10	0.2
11	0.232
12	0.2
13	0.256
14	0.196
15	0.188

Dataset: 300

Pruned Tree	Misclassification Error rate
16	0.263
17	0.293
18	0.3
19	0.293
20	0.233

Dataset:500

21	0.294
22	0.286
23	0.294
25	0.274
26	0.25
27	0.284
28	0.29
29	0.256
30	0.274

2. Average Error Rates on 10 Datasets

# of Pruned tree	Average Error rate	**Error rates on different datasets**									
		1000	900	800	700	600	500	400	300	200	100
19	0.2735	0.276	0.2756	0.2875	0.2671	0.2717	0.2680	0.2825	0.2967	0.2300	0.2800
20	0.2509	0.253	0.2544	0.2575	0.2443	0.2417	0.2400	0.2700	0.2733	0.2350	0.2400
3	0.2533	0.254	0.2567	0.2513	0.2500	0.2317	0.2440	0.2875	0.2633	0.2650	0.2300
4	0.2490	0.246	0.2411	0.2500	0.2400	0.2450	0.2400	0.2475	0.2600	0.2300	0.2900
5	0.2752	0.271	0.2700	0.2675	0.2629	0.2517	0.2740	0.3000	0.2900	0.2850	0.2800
10	0.2561	0.256	0.2622	0.2488	0.2371	0.2683	0.266	0.2375	0.3000	0.285	0.2
13	0.2581	0.259	0.2656	0.2538	0.2386	0.2833	0.272	0.2225	0.3067	0.28	0.2
14	0.2578	0.258	0.2633	0.2538	0.2386	0.2817	0.272	0.2225	0.3033	0.275	0.21
11	0.2631	0.263	0.2678	0.2513	0.2486	0.2750	0.254	0.2450	0.2967	0.31	0.2200
8	0.2481	0.25	0.2522	0.2538	0.2557	0.2517	0.268	0.2475	0.2367	0.235	0.2300
27	0.2355	0.242	0.2467	0.2513	0.2314	0.2350	0.224	0.2525	0.2667	0.205	0.2
23	0.2613	0.265	0.2622	0.2713	0.2571	0.2533	0.248	0.2825	0.2833	0.24	0.25
21	0.2652	0.269	0.2722	0.2750	0.2557	0.2583	0.252	0.285	0.3000	0.245	0.24
22	0.2568	0.258	0.2622	0.2650	0.2443	0.2450	0.236	0.2775	0.2900	0.23	0.26
30	0.2670	0.271	0.2700	0.2863	0.2600	0.2667	0.252	0.2775	0.2967	0.21	0.28
25	0.2478	0.253	0.2522	0.2713	0.2429	0.2483	0.234	0.26	0.2767	0.18	0.26
26	0.2569	0.256	0.2578	0.2600	0.2486	0.2450	0.246	0.2725	0.2733	0.24	0.27
29	0.2490	0.255	0.2611	0.2613	0.2414	0.2450	0.24	0.27	0.2867	0.23	0.2